



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/943,900	08/30/2001	Yan J. Arrouye	P2209USC1	2367

7590 08/23/2005

APPLE COMPUTER, INC.  
1 INFINITE LOOP, M/S 3-PAT  
CUPERTINO, CA 95014

EXAMINER
----------

BULLOCK JR, LEWIS ALEXANDER

ART UNIT	PAPER NUMBER
----------	--------------

2195

DATE MAILED: 08/23/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary

Application No.

09/943,900

Applicant(s)

ARROUYE ET AL.

Examiner

Lewis A. Bullock, Jr.

Art Unit

2195

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 31 May 2005.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 24-37 and 52-75 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 24-37 and 52-75 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 04 February 2002 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some \* c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
- \* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- 1) ☒ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)  
Paper No(s)/Mail Date \_\_\_\_\_
- 4) ☐ Interview Summary (PTO-413)  
Paper No(s)/Mail Date. \_\_\_\_\_
- 5) ☐ Notice of Informal Patent Application (PTO-152)
- 6) ☐ Other: \_\_\_\_\_

## **DETAILED ACTION**

### ***Terminal Disclaimer***

1. The terminal disclaimer filed on May 31, 2005 disclaiming the terminal portion of any patent granted on this application which would extend beyond the expiration date of U.S. Patent 5,615,400 has been reviewed and is accepted. The terminal disclaimer has been recorded.

### ***Claim Rejections - 35 USC § 112***

2. The following is a quotation of the first paragraph of 35 U.S.C. 112:

The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

3. Claim 25 is rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the written description requirement. The claim(s) contains subject matter which was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the inventor(s), at the time the application was filed, had possession of the claimed invention. The cited claim details, "the software library is scheduled to be removed before the execution of the library routine is completed." The examiner, at best, finds support for the software library is scheduled to be removed before the execution of the library routine is performed such that the scheduled removal is canceled and the library routine is executed. The examiner finds no support in the specification for scheduling the removal of the library before the execution of the library

routine is completed, which when interpreted would mean that the removal is scheduled while library routine is executing.

***Claim Rejections - 35 USC § 103***

4. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

5. Claims 24-37 and 52-75 are rejected under 35 U.S.C. 103(a) as being unpatentable over CROWSAR (U.S. Patent 5,615,400) in view of THATTE (U.S. patent 4,695,949).

As to claim 27, CROWSAR teaches a computer system for dynamically and automatically loading and unloading a software library (libraries) to and from memory in a computer, the software library including one or more library routines (functions) and capable of being referenced by an application or other software module (col. 4, lines 33-45; col. 4, lines 55-56; col. 13, line 64 – col. 14, line 23), the computer system comprising:

one or more library structures (libraries), each library structure corresponding to a software library, each library structure including a library implementation module (constructor / code resources / implementation segment) containing code for implementing the corresponding software library (col. 9, lines 49 – col. 10, line 18; col. 14, line 65 – col. 14, line 3; col. 15, lines 15-19) and a library loader (jump table)

containing entry points corresponding to entry points of the corresponding software library and code for loading and unloading the corresponding library implementation module (via the jump table containing pc-relative jsr instructions that goes to the SLM segment loader entry point...) (col. 15, lines 20-61); and software to perform operations comprising: loading into memory a library implementation module (implementation segment) for a software library, said loading occurring prior to when a library routine of the software library is to be executed by the application or other software module (via on-demand loading) (col. 13, line 64 – col. 14, line 6; col. 14, lines 15-23; col. 14, line 65 – col. 14, line 3; col. 15, lines 15-23), and that shared libraries are dynamically loaded and unloaded based on use counts (col. 14, lines 15-23) wherein the load function increments the use count and after access of the library the use count is decremented (col. 11, lines 11-19; col. 9, lines 35-39) such that the memory management system can unload libraries which are not in current use by an active application (col. 4, lines 42-45). It is inherent with the teachings of COWSAR that since the use count is used to determine accessing and loading of a library that if the library is in memory its use-count is incremented and if it is not in memory it is loaded and the use-count is subsequently incremented as well as the library being in memory and not be used, i.e. having a zero use count. However, COWSAR does not explicitly teach if the library implementation module is not in memory and scheduled to be unloaded from memory, then canceling the scheduled unloading of the library implementation module; and automatically scheduling the unloading of the library implementation module; and unless the scheduled unload has been cancelled automatically unloading the library

implementation module as scheduled wherein the software library is scheduled for unload and unloaded automatically without an explicit unload request from the application or software module.

THATTE teaches memory management technique wherein if a referenced module (memory block) is not in memory and scheduled to be unloaded from memory (i.e. the memory block has a zero reference count), then canceling the scheduled unloading of the module (increasing the reference count for the memory block and removing it from the reference count filter) (col. 7, lines 27-39; col. 8, lines 27-38); and automatically scheduling the unloading of the module (via placing the memory block into the reference count filter to be sent to the garbage buffer); and unless the scheduled unload has been cancelled (via a subsequent reference to increase the reference count from zero to one) (col. 7, lines 27-39; col. 8, lines 27-38) automatically unloading the module as scheduled wherein the module is scheduled for unload and unloaded automatically without an explicit unload request from the application or software module (via a background process that garbage the memory blocks that are not referenced or subsequently referenced) (col. 9, lines 1-25). It would be obvious to one skilled in the art at the time of the invention that since the libraries determine access, i.e. reference based upon use counts, that the use counts are the referenced counts and the libraries are the memory blocks based upon the combination. Therefore, it would be obvious to one skilled in the art at the time of the invention to combine the teachings of COWSAR with the teachings of THATTE in order to efficient, cost-effective method of alleviating

the need for frequent garbage collection, reduces the reference counting overhead, and allows reference counting to be implemented in practice (col. 5, lines 23-28).

As to claim 24, refer to claim 27 for rejection. However, claim 24 further details determining which libraries are potentially needed during execution of an application; and loading those libraries via the library loader and implementation modules. COWSAR teaches that libraries are on-demand loaded into the system based upon the application, therefore they are referenced and essentially needed by the application (col. 13, line 64 – col. 14, line 6; col. 14, lines 15-23; col. 14, line 65 – col. 14, line 3; col. 15, lines 15-23). COWSAR also teaches loading libraries that have library loaders (jump tables) and implementation modules (implementation segment) (col. 9, lines 49 – col. 10, line 18; col. 14, line 65 – col. 14, line 3; col. 15, lines 15-19) (via the jump table containing pc-relative jsr instructions that goes to the SLM segment loader entry point...) (col. 15, lines 20-61).

As to claim 36, refer to claim 27 for rejection.

As to claims 52-54, refer to claim 27 for rejection. However, claim 52 further details the unloading of the library implementation module includes setting a time delay and unloading the library implementation module if after the time delay has been met the library implementing module is not in use. COWSAR teaches that libraries are referenced by use counts and include implementation modules (col. 9, lines 49 – col.

10, line 18; col. 14, line 65 – col. 14, line 3; col. 15, lines 15-19). THATTE teaches that unreferenced modules are referenced, unreferenced, rereferenced or reclaimed by a memory management system when their reference count is zero (col. 8, lines 1-17). Hence, when an unreferenced module (a memory block having a zero reference count) is re-referenced, the module increments its reference counter from zero to one before it is garbage collected by a background process. Therefore, it would be obvious to one skilled in the art that the garbage collection is performed separately from the handling of the reference counter. “Official Notice” is taken in that it is well known in the art that a user or system has a set time of operation or criteria, i.e. low memory, for performing garbage collection or memory management operations on a computer system and therefore would be obvious in view of COWSAR and THATTE in order to set a time delay or criteria for officially unloading a library, i.e. run the garbage collection algorithm, when the library is not in use.

As to claim 55, refer to claim 52 for rejection.

As to claim 56, refer to claim 52 for rejection.

As to claim 57, refer to claim 52 for rejection.

As to claim 58, refer to claim 52 for rejection.



As to claim 59, refer to claim 24 for rejection.

As to claim 60-62, refer to claim 24 for rejection. However, claim 60 further details the unloading of the library implementation module includes setting a time delay and unloading the library implementation module if after the time delay has been met the library implementing module is not in use. COWSAR teaches that libraries are referenced by use counts and include implementation modules (col. 9, lines 49 – col. 10, line 18; col. 14, line 65 – col. 14, line 3; col. 15, lines 15-19). THATTE teaches that unreferenced modules are referenced, unreferenced, rereferenced or reclaimed by a memory management system when their reference count is zero (col. 8, lines 1-17). Hence, when an unreferenced module (a memory block having a zero reference count) is re-referenced, the module increments its reference counter from zero to one before it is garbage collected by a background process. Therefore, it would be obvious to one skilled in the art that the garbage collection is performed separately from the handling of the reference counter. “Official Notice” is taken in that it is well known in the art that a user or system has a set time of operation or criteria, i.e. low memory, for performing garbage collection or memory management operations on a computer system and therefore would be obvious in view of COWSAR and THATTE in order to set a time delay or criteria for officially unloading a library, i.e. run the garbage collection algorithm, when the library is not in use.

As to claim 63, refer to claim 24 for rejection.

As to claims 64-67, refer to claims 60-62 for rejection.

As to claims 68-73, refer to claims 60-62 for rejection. However, claim 68 further details the system comprising a processor; a memory; and a disk. COWSAR teaches the disclosed limitations (col. 6, lines 19-33).

As to claims 74, refer to claim 60 for rejection.

As to claim 75, refer to claim 60 for rejection.

As to claims 25 and 26, COWSAR teaches loading of library is performed after execution of the application begins but before the library routine is executed (on-demand loading) (libraries are loaded at any time) (col. 13, line 64 – col. 14, line 23).

As to claims 28 and 29, refer to claims 25 and 26 for rejection.

As to claims 30-35, COWSAR teaches that libraries are referenced by use counts and include implementation modules (col. 9, lines 49 – col. 10, line 18; col. 14, line 65 – col. 14, line 3; col. 15, lines 15-19). THATTE teaches that unreferenced modules are referenced, unreferenced, rereferenced or reclaimed by a memory management system when their reference count is zero (col. 8, lines 1-17). Hence,

when an unreferenced module (a memory block having a zero reference count) is re-referenced, the module increments its reference counter from zero to one before it is garbage collected by a background process. Therefore, it would be obvious to one skilled in the art that the garbage collection is performed separately from the handling of the reference counter. "Official Notice" is taken in that it is well known in the art that a user or system has a set time of operation or criteria, i.e. low memory, for performing garbage collection or memory management operations on a computer system and therefore would be obvious in view of COWSAR and THATTE in order to set a time delay or criteria for officially unloading a library, i.e. run the garbage collection algorithm, when the library is not in use.

As to claim 37, refer to claims 29-35 for rejection.

### ***Response to Arguments***

6. Applicant's arguments filed May 31, 2005 have been fully considered but they are not persuasive. Applicant argues that the examiner states that Cowser does not teach "automatically scheduling the unloading of the library implementation module". The examiner disagrees. What the examiner states that is not explicitly taught by Cowser is the canceling of the scheduled unloading. As defined in the specification at page 13, lines 1-2, "If the use count equals zero and the library is loaded in memory, then the library is considered to be scheduled for unload." Therefore, the fact that the use count is zero, is the scheduling for unloading. Cowser teaches a use count for library

structures that represents who is referencing the associated library structure wherein the value is zero indicates no references or a non-zero value indicates one or more references. Therefore, Cowser teaches the scheduling for unloading by having a value of zero. Cowser also teaches accessing a memory structure by incrementing its use count. Thatte teaches a reference counting technique wherein memory blocks that have a zero value reference counter are initially stored in a virtual list such that after a predetermined time if the value is still zero the memory block is reclaimed by a memory management structure known as a garbage collector. It would be obvious to one skilled in the art at the time of the invention that since the use count of Cowser is a reference counter to the library than it is the memory block having a reference counter as defined by Thatte and therefore the actual reclamation of the memory is delayed and subsequently performed if the reference count is still zero. Therefore, the examiner provides this reasoning in response to Applicant's current and subsequent arguments regarding the teachings of Thatte in combination with Cowser regarding all limitations of automatically scheduling of unloading of the library / library implementation module after the execution of the library routine is completed by performing or not performing a canceling operation. Therefore, since the combination still teaches the limitations as disclosed the rejection is maintained.

***Conclusion***

7. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).


A shortened statutory period for reply to this final action is set to expire **THREE MONTHS** from the mailing date of this action. In the event a first reply is filed within **TWO MONTHS** of the mailing date of this final action and the advisory action is not mailed until after the end of the **THREE-MONTH** shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than **SIX MONTHS** from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Lewis A. Bullock, Jr. whose telephone number is (571) 272-3759. The examiner can normally be reached on Monday-Friday, 8:30 - 5:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Meng An can be reached on (571) 272-3756. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

August 19, 2005



**LEWIS A. BULLOCK, JR.**  
**PRIMARY EXAMINER**